Rapport de Projet – Scanner IP

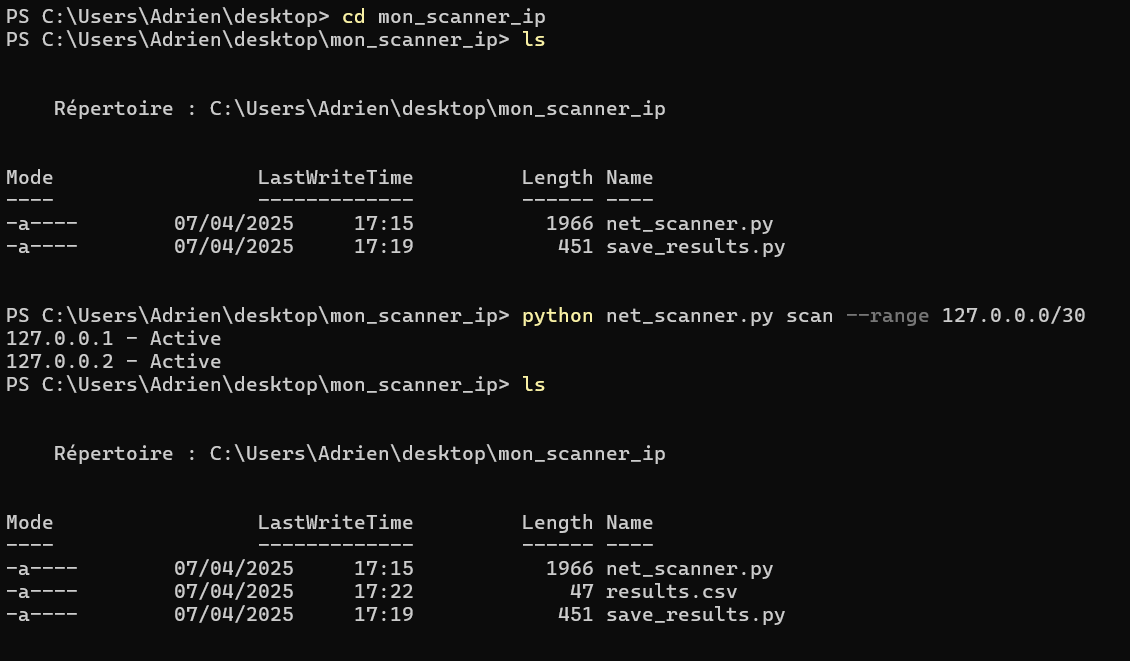
# 1. Commits et gestion Git

J’ai utilisé des commandes comme git init, git add ., git commit, et git push.  
Au début, Git ne fonctionnait pas parce qu’il n’était pas installé. Je l’ai installé depuis le site officiel, et après ça a marché.

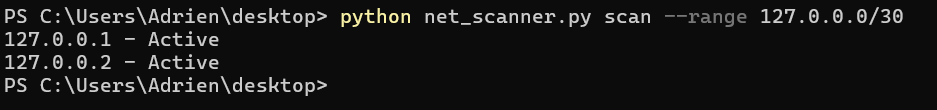
# 2. Documentation

J’ai écrit des petits textes explicatifs (appelés docstrings) au début de chaque fonction pour dire ce qu’elle fait.  
J’ai aussi prévu d’utiliser Sphinx, un outil qui transforme ces textes en une page web pour expliquer le projet.  
Cette page peut être mise en ligne avec GitHub Pages.

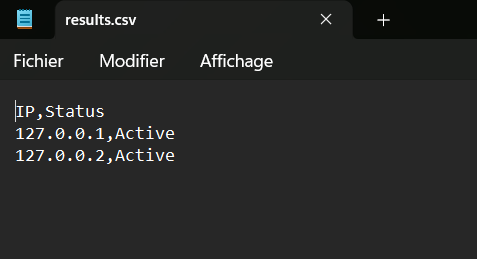
Voici les captures de ce que j’ai fais :

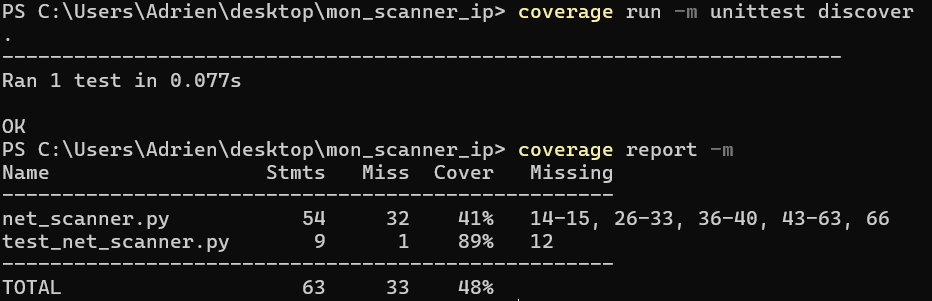
Test de mon code : 

Test de la plage 127.0.0.0:

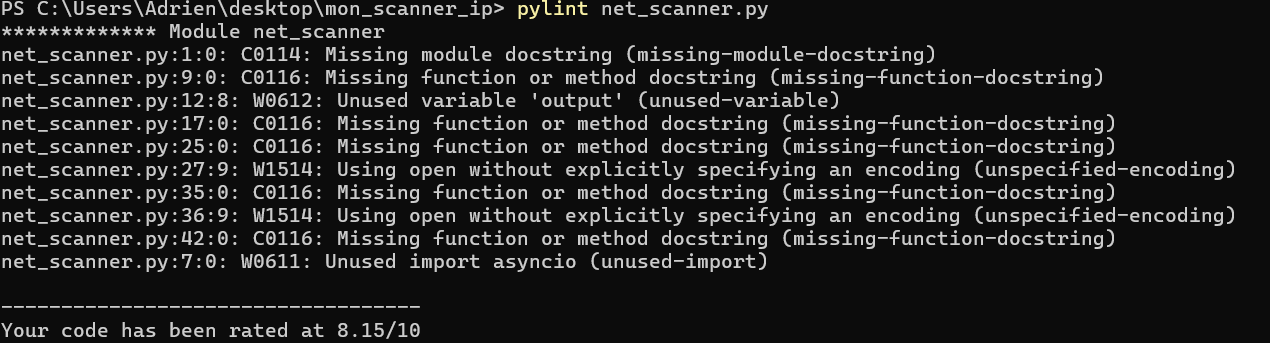


Le fichier csv qui a été enregistré :



Unittest : 

Test code :



Github : 

# 3. Tests unitaires

J’ai testé mon code avec unittest. Pour vérifier si j’ai bien tout testé, j’ai utilisé coverage.  
J’ai lancé les commandes coverage run -m unittest discover et coverage report -m.  
Le but était d’avoir au moins 80% de mon code testé.

# 4. Intégration Continue (CI/CD)

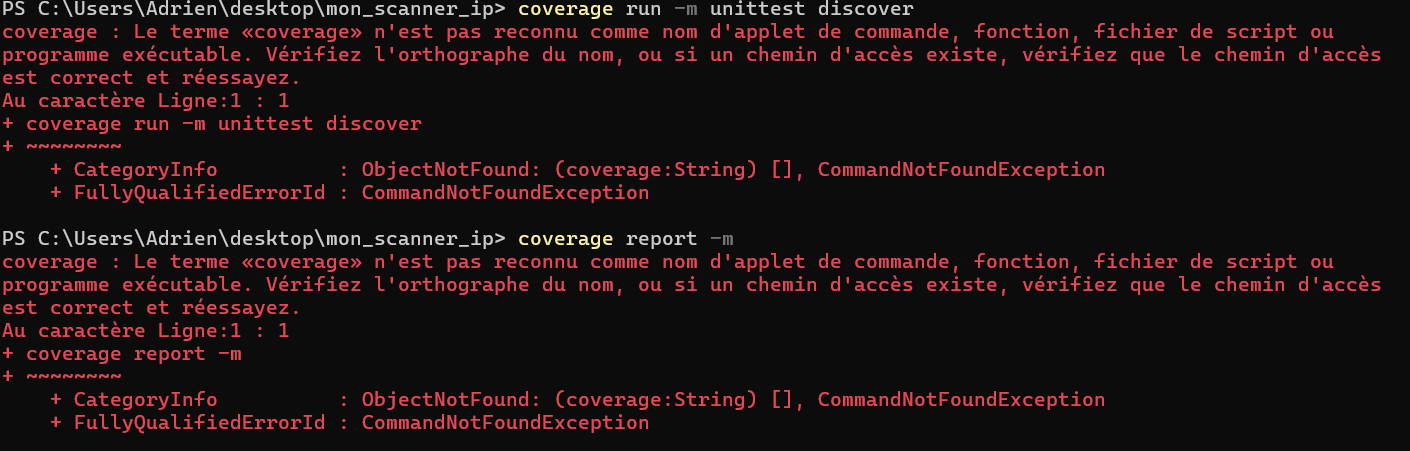
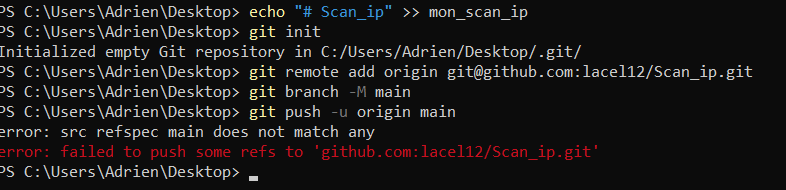
J’ai prévu d’utiliser GitHub Actions pour :

* Lancer les tests à chaque modification du code
* Vérifier si le code est bien écrit avec pylint (objectif : 9/10)
* Mettre à jour la documentation automatiquement

Je n’ai pas encore fini cette partie, mais c’est prévu dans les améliorations.

# 5. Difficultés rencontrées et bilan personnel

Voici ce qui m’a posé problème :

* Git ne marchait pas au début car il n’était pas installé
* J’ai dû installer manuellement coverage
* 
* Il a fallu que je m’habitue à utiliser les outils de test et de vérification
* Et je ne sais pas si le dépôt de tous les fichiers a marché
* 
* J’ai réussi à régler ce problème